Distributed Quantum Proofs for Replicated Data*

Pierre Fraigniaud
CNRS and Université de Paris

François Le Gall
Nagoya University

Harumichi Nishimura
Nagoya University

Ami Paz
Universität Wien

Abstract

This paper tackles the issue of *checking* that all copies of a large data set replicated at several nodes of a network are identical. The fact that the replicas may be located at distant nodes prevents the system from verifying their equality locally, i.e., by having each node consult only nodes in its vicinity. On the other hand, it remains possible to assign *certificates* to the nodes, so that verifying the consistency of the replicas can be achieved locally. However, we show that, as the replicated data is large, classical certification mechanisms, including distributed Merlin-Arthur protocols, cannot guarantee good completeness and soundness simultaneously, unless they use very large certificates. The main result of this paper is a distributed *quantum* Merlin-Arthur protocol enabling the nodes to collectively check the consistency of the replicas, based on small certificates, and in a single round of message exchange between neighbors, with short messages. In particular, the certificate-size is logarithmic in the size of the data set, which gives an exponential advantage over classical certification mechanisms. We propose yet another usage of a fundamental quantum primitive, called the SWAP test, in order to show our main result.

Technical version available at arXiv:2002.10018.

Background In the context of distributed systems, the presence of faults potentially corrupting the individual states of the nodes creates a need to regularly check whether the system is in a global state that is legal with respect to its specification. A basic example is a system storing data, and using replicas in order to support crash failures. In this case, the application managing the data is in charge of regularly checking that the several replicas of the same data, stored at different nodes scattered in the network, are all identical. Another example is an application maintaining a tree spanning the nodes of a network, e.g., for multicast communication. In this case, every node stores a pointer to its parent in the tree, and the application must regularly check that the collection of pointers forms a spanning tree. This paper addresses the issue of checking the correctness of a distributed system configuration at low cost.

Several mechanisms have been designed for certifying the correctness of the global state of a system in a distributed manner. One popular mechanism is called *locally checkable proofs* [22], and it extends the seminal concept of *proof-labeling schemes* [32]. In these frameworks, the distributed application does not only construct or maintain some distributed data structure (e.g., a spanning tree), but also constructs a distributed *proof* that the data structure is correct. This proof has the form of a *certificate* assigned to each node (the certificates assigned to different nodes do not need to be the same). For collectively checking the legality of the current global system state, the nodes exchange their certificates with their neighbors in the network. Then, based on its own individual state, its certificate, and the certificates of its neighbors, every node accepts or rejects, according to the following specification. If the global state is legal, and if the certificates are assigned properly by the application, then all nodes accept. Conversely, if the global state is illegal, then at least one node rejects, *no matter which certificates*

*A conference version of this work will be presented in the 12th Innovations in Theoretical Computer Science Conference (ITCS2021).

are assigned to the nodes. Such a rejecting node can raise an alarm, or launch a recovery procedure. The main aim of locally checkable proofs is to be *compact*, that is, to use certificates as small as possible, for two reasons: first, to limit the space complexity at each node, and, second, to limit the message complexity of the verification procedure involving communications between neighbors.

Unfortunately, not all boolean predicates on labeled graphs can be distributedly certified using certificates as small as for spanning tree. This is typically the case of the aforementioned scenario of a distributed data storage using replicas, for which one must certify equality. Let us for instance consider the case of two nodes Alice and Bob at the two extremities of a path, that is, the two players are separated by intermediate nodes. Alice and Bob respectively store two n-bit strings x and y, and the objective is to certify that $x = y$. That is, one wants to certify equality (EQ) between *distant* players. A direct reduction from the non-deterministic communication complexity of EQ shows that certifying EQ cannot be achieved with certificates smaller than $\Omega(n)$ bits.

Randomization may help circumventing the difficulty of certifying some boolean predicates on labeled graphs using small certificates. Hence, a weaker form of protocols has been considered, namely *distributed Merlin-Arthur* protocols (dMA), a.k.a. *randomized proof-labeling schemes* [20]. In this latter context, Merlin provides the nodes with a proof, just like in locally checkable proofs, and Arthur performs a *randomized* local verification at each node. Unfortunately, some predicates remain hard in this framework too. In particular, as we show in the paper, there are no classical dMA protocols for (distant) EQ using compact certificates. Recently, several extensions of dMA protocols were proposed, e.g., by allowing more interaction between the prover and the verifier [14, 19, 35]. In this work, we add the quantum aspect, while considering only a single interaction, and only in the prescribed order: Merlin sends a proof to Arthur, and then there is no more interaction between them.

Our Results We carry on the recent trend of research consisting of investigating the power of quantum resources in the context of distributed network computing (cf., e.g., [16, 33, 25, 34, 26, 21]), by designing a distributed Quantum Merlin-Arthur (dQMA) protocol for distant EQ, using compact certificates and small messages. While we use the dQMA terminology in order to be consistent with prior work, we emphasize that the structure of the discussed protocols is rather simple: each node is given a quantum state as a certificate, the nodes exchange these states, perform a local computation, and finally accept or reject.

Our main result is the following. A collection of n-bit strings $x_1, \ldots, x_t$ are stored at t terminal nodes $u_1, \ldots, u_t$ in a network $G = (V, E)$, where node u_i stores x_i. We denote EQ_n^t the problem of checking the equality $x_1 = \cdots = x_t$ between the t strings. Let us define the *radius* of a given instance of EQ_n^t as $r = \min_i \max_j \mathsf{dist}_G(u_i, u_j)$, where dist_G denotes the distance in the (unweighted) graph G. Our main result is the design of a dQMA protocol for EQ_n^t, using small certificate. This can be summarized by the following informal statement (the formal statement is in the technical version):

Theorem 1. *There is a distributed Quantum Merlin-Arthur (dQMA) protocol for certifying equality between t binary strings (EQ_n^t) of length n, and located at a radius-r set of t terminals, in a single round of communication between neighboring nodes using certificates of size $O(tr^2 \log n)$ qubits, and messages of size $O(tr^2 \log(n + r))$ qubits.*

It is worth mentioning that, although the dependence in r and t is polynomial, the dependence in the actual size n of the instance remains logarithmic, which is our main concern. Indeed, for applications such as the aforementioned distributed data storage motivating the distant EQ_n^t problem, it is expected that both the number t of replicas, and the maximum distance between the nodes storing these replicas are of several orders of magnitude smaller than the size n of the stored replicated data.

It is also important to note that our protocol satisfies the basic requirement of *reusability*, as one aims for protocols enabling regular and frequent verifications that the data are not corrupted. Specifically, the quantum operations performed on the certificates during the local verification phase operated between neighboring nodes preserve the quantum nature of these certificates. That is, if EQ_n^t is satisfied, i.e., if all the replicas x_i's are equal, then, up to an elementary local relocation of the quantum certificates, these certificates are available for a next

test. If EQ_n^t is not satisfied, i.e., if there exists a pair of replicas $x_i \neq x_j$, then the certificates do not need to be preserved as this scenario corresponds to the case where the correctness of the data structure is violated, requiring the activation of recovery procedures for fixing the bug, and reassigning certificates to the nodes.

Our quantum protocol is based on the SWAP test [11], which is a basic tool in the theory of quantum computation and quantum information. This test allows to check if a quantum state is symmetric, and has several applications, such as estimating the inner product of two states (e.g., [11, 8, 39]), checking whether a given state (or a reduced state of it) is pure or entangled with the environment system (e.g., [1, 30, 23, 29]), and more. In this paper, we use the SWAP test in yet another way: *for checking if two of the reduced states of a given state are close*. A similar use was done by [37] in a different context (transforming quantum circuits to shallow ones in a hardness reduction proof).

Finally, observe that our logarithmic upper bound for dQMA protocols is in contrast to the linear lower bound that can be shown for classical dMA protocols even for $t = 2$ on a path of 4 nodes and even for the case where communication between the neighboring nodes is extended to multiple rounds (see precise statement and proof in the technical version). Our results thus show that quantum certification mechanism can provide an exponential advantage over classical certification mechanisms.

Related Work The concept of distributed proofs is a part of the framework of distributed network computing since the early works on fault-tolerance (see, e.g., [2, 5, 24]). Proof-labeling schemes were introduced in [32], and variants have been studied in [22, 18]. Randomized proof-labeling schemes have been studied in [20]. Extensions of distributed proofs to a hierarchy of decision mechanisms have been studied in [17] and [6]. Frameworks like cloud computing recently enabled envisioning systems in which the nodes of the network could interact with a third party, leading to the concept of *distributed interactive proofs* [31]. There, each node can interact with an *oracle* who has a complete view of the system, is computationally unbounded, but is not trustable. For instance, in Arthur-Merlin (dAM) protocols, the nodes start by querying the oracle Merlin, which provides them with answers in their certificates. There is a simple classical compact dAM protocol for distant EQ, where the two players stand at the extremities of a path. We refer to [14, 19, 35] for recent developments in the framework of distributed interactive proofs. While distributed Arthur-Merlin protocols and their extensions provide an appealing theoretical framework for studying the power of interactive proofs in the distributed setting, the practical implementation of such protocols remains questionable, since they all require the existence of a know-all oracle, Merlin, and it is unclear if a Cloud could play this role. On the other hand, in dMA and dQMA protocols, interaction with an external party is not required, but only a one-time assignment of certificates is needed, which are then reusable for regular verification. As in the classical proof-labeling schemes setting, these certificates can actually be *created* by the nodes themselves during a pre-processing phase, making the reliance on a know-all oracle unnecessary.

After a few early works [7, 16, 21, 38] that shed light on the potential and limitations of quantum distributed computing (see also [4, 10, 15] for general discussions), evidence of the advantage of quantum distributed computing over classical distributed computing have been obtained recently for three fundamental models of (synchronous fault-free) distributed network computing: the CONGEST model [26, 33], the CONGEST-CLIQUE model [25] and the LOCAL model [34]. The present paper adds to this list another important task for which quantum distributed computing significantly outperforms classical distributed computing, namely, distributed certification.

Note that while this paper is the first to study quantum Merlin-Arthur protocols in a distributed computing framework, there are a number of prior works studying them in communication complexity [36, 27, 28, 9]. In particular, quantum Merlin-Arthur protocols are shown to improve some computational measure (say, the total length of the messages from the prover to Alice, and of the messages between Alice and Bob) exponentially compared to Merlin-Arthur protocols where the messages from the prover are classical [36, 28].

The question of computing functions on inputs that are given to graph nodes was also studied in the context of communication complexity. The equality function was studied for the case where all nodes have inputs [3]. Other works considered a setting similar to ours, i.e., where only some nodes have inputs [12, 13], but did not study the equality problem.

References

[1] Scott Aaronson, Salman Beigi, Andrew Drucker, Bill Fefferman, and Peter W. Shor. The power of unentanglement. *Theory of Computing*, 5:1:1–1:42, 2009. doi:10.4086/toc.2009.v005a001.

[2] Yehuda Afek, Shay Kutten, and Moti Yung. The local detection paradigm and its application to self-stabilization. *Theoretical Computer Science*, 186(1-2):199–229, 1997. doi:10.1016/S0304-3975(96)00286-1.

[3] Noga Alon, Klim Efremenko, and Benny Sudakov. Testing equality in communication graphs. *IEEE Transactions on Information Theory*, 63(11):7569–7574, 2017. doi:10.1109/TIT.2017.2744608.

[4] Heger Arfaoui and Pierre Fraigniaud. What can be computed without communications? *SIGACT News*, 45(3):82–104, 2014. doi:10.1145/2670418.2670440.

[5] Baruch Awerbuch, Boaz Patt-Shamir, and George Varghese. Self-stabilization by local checking and correction (extended abstract). In *32nd Symposium on Foundations of Computer Science (FOCS)*, pages 268–277, 1991. doi:10.1109/SFCS.1991.185378.

[6] Alkida Balliu, Gianlorenzo D'Angelo, Pierre Fraigniaud, and Dennis Olivetti. What can be verified locally? *Journal of Computer System and Sciences*, 97:106–120, 2018. doi:10.1016/j.jcss.2018.05.004.

[7] Michael Ben-Or and Avinatan Hassidim. Fast quantum byzantine agreement. In *37th Annual ACM Symposium on Theory of Computing (STOC)*, pages 481–485, 2005. doi:10.1145/1060590.1060662.

[8] Hugue Blier and Alain Tapp. A quantum characterization of NP. *Computational Complexity*, 21(3):499–510, 2012. doi:10.1007/s00037-011-0016-2.

[9] Ralph Bottesch, Dmitry Gavinsky, and Hartmut Klauck. Equality, revisited. In *40th International Symposium on Mathematical Foundations of Computer Science (MFCS)*, volume 9235 of *LNCS*, pages 127–138. Springer, 2015. doi:10.1007/978-3-662-48054-0_11.

[10] Anne Broadbent and Alain Tapp. Can quantum mechanics help distributed computing? *SIGACT News*, 39(3):67–76, 2008. doi:10.1145/1412700.1412717.

[11] Harry Buhrman, Richard Cleve, John Watrous, and Ronald de Wolf. Quantum fingerprinting. *Physical Review Letters*, 87(16):167902:1–167902:4, 2001. doi:10.1103/PhysRevLett.87.167902.

[12] Arkadev Chattopadhyay, Jaikumar Radhakrishnan, and Atri Rudra. Topology matters in communication. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS*, pages 631–640, 2014. doi:10.1109/FOCS.2014.73.

[13] Arkadev Chattopadhyay and Atri Rudra. The range of topological effects on communication. In *Automata, Languages, and Programming - 42nd International Colloquium, ICALP*, pages 540–551, 2015. doi:10.1007/978-3-662-47666-6_43.

[14] Pierluigi Crescenzi, Pierre Fraigniaud, and Ami Paz. Trade-offs in distributed interactive proofs. In *33rd International Symposium on Distributed Computing (DISC)*, pages 13:1–13:17, 2019. doi:10.4230/LIPIcs.DISC.2019.13.

[15] Vasil S. Denchev and Gopal Pandurangan. Distributed quantum computing: a new frontier in distributed systems or science fiction? *SIGACT News*, 39(3):77–95, 2008. doi:10.1145/1412700.1412718.

[16] Michael Elkin, Hartmut Klauck, Danupon Nanongkai, and Gopal Pandurangan. Can quantum communication speed up distributed computation? In *33rd ACM Symposium on Principles of Distributed Computing (PODC)*, pages 166–175, 2014. doi:10.1145/2611462.2611488.

[17] Laurent Feuilloley, Pierre Fraigniaud, and Juho Hirvonen. A hierarchy of local decision. In *43rd International Colloquium on Automata, Languages, and Programming (ICALP)*, pages 118:1–118:15, 2016. doi:10.4230/LIPIcs.ICALP.2016.118.

[18] Pierre Fraigniaud, Amos Korman, and David Peleg. Towards a complexity theory for local distributed computing. *Journal of the ACM*, 60(5):35:1–35:26, 2013. doi:10.1145/2499228.

[19] Pierre Fraigniaud, Pedro Montealegre, Rotem Oshman, Ivan Rapaport, and Ioan Todinca. On distributed Merlin-Arthur decision protocols. In *26th International Colloquium on Structural Information and Commu-*

nication Complexity (SIROCCO), volume 11639 of *LNCS*, pages 230–245. Springer, 2019. doi:10.1007/978-3-030-24922-9_16.

[20] Pierre Fraigniaud, Boaz Patt-Shamir, and Mor Perry. Randomized proof-labeling schemes. *Distributed Computing*, 32(3):217–234, 2019. doi:10.1007/s00446-018-0340-8.

[21] Cyril Gavoille, Adrian Kosowski, and Marcin Markiewicz. What can be observed locally? In *23rd International Symposium on Distributed Computing (DISC)*, volume 5805 of *LNCS*, pages 243–257. Springer, 2009. doi:10.1007/978-3-642-04355-0_26.

[22] Mika Göös and Jukka Suomela. Locally checkable proofs in distributed computing. *Theory of Computing*, 12:19:1–19:33, 2016. doi:10.4086/toc.2016.v012a019.

[23] Aram W. Harrow and Ashley Montanaro. Testing product states, quantum Merlin-Arthur games and tensor optimization. *Journal of the ACM*, 60(1):3:1–3:43, 2013. doi:10.1145/2432622.2432625.

[24] Gene Itkis and Leonid A. Levin. Fast and lean self-stabilizing asynchronous protocols. In *35th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 226–239, 1994. doi:10.1109/SFCS.1994.365691.

[25] Taisuke Izumi and François Le Gall. Quantum distributed algorithm for the All-Pairs Shortest Path problem in the CONGEST-CLIQUE model. In *38th ACM Symposium on Principles of Distributed Computing (PODC)*, pages 84–93, 2019. doi:10.1145/3293611.3331628.

[26] Taisuke Izumi, François Le Gall, and Frédéric Magniez. Quantum distributed algorithm for triangle finding in the CONGEST model. In *37th International Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 23:1–23:13, 2020. doi:10.4230/LIPIcs.STACS.2020.23.

[27] Hartmut Klauck. On Arthur Merlin games in communication complexity. In *26th Annual IEEE Conference on Computational Complexity (CCC)*, pages 189–199, 2011. doi:10.1109/CCC.2011.33.

[28] Hartmut Klauck and Supartha Podder. Two results about quantum messages. In *39th International Symposium on Mathematical Foundations of Computer Science (MFCS)*, volume 8635 of *LNCS*, pages 445–456. Springer, 2014. doi:10.1007/978-3-662-44465-8_38.

[29] Hirotada Kobayashi, François Le Gall, and Harumichi Nishimura. Stronger methods of making quantum interactive proofs perfectly complete. *SIAM Journal on Computing*, 44(2):243–289, 2015. doi:10.1137/140971944.

[30] Hirotada Kobayashi, Keiji Matsumoto, and Tomoyuki Yamakami. Quantum Merlin-Arthur proof systems: Are multiple Merlins more helpful to Arthur? *Chicago Journal of Theoretical Computer Science*, 2009:3:1–3:19, 2009. doi:10.4086/cjtcs.2009.003.

[31] Gillat Kol, Rotem Oshman, and Raghuvansh R. Saxena. Interactive distributed proofs. In *37th ACM Symposium on Principles of Distributed Computing (PODC)*, pages 255–264, 2018. doi:10.1145/3212734.3212771.

[32] Amos Korman, Shay Kutten, and David Peleg. Proof labeling schemes. *Distributed Computing*, 22(4):215–233, 2010. doi:10.1007/s00446-010-0095-3.

[33] François Le Gall and Frédéric Magniez. Sublinear-time quantum computation of the diameter in CONGEST networks. In *37th ACM Symposium on Principles of Distributed Computing (PODC)*, pages 337–346, 2018. doi:10.1145/3212734.3212744.

[34] François Le Gall, Harumichi Nishimura, and Ansis Rosmanis. Quantum advantage for the LOCAL model in distributed computing. In *36th International Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 49:1–49:14, 2019. doi:10.4230/LIPIcs.STACS.2019.49.

[35] Moni Naor, Merav Parter, and Eylon Yogev. The power of distributed verifiers in interactive proofs. In *31st ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1096–115, 2020. doi:10.1137/1.9781611975994.67.

[36] Ran Raz and Amir Shpilka. On the power of quantum proofs. In *19th IEEE Conference on Computational Complexity (CCC)*, pages 260–274, 2004. doi:10.1109/CCC.2004.1313849.

[37] Bill Rosgen. Distinguishing short quantum computations. In *25th International Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 597–608, 2008. doi:10.4230/LIPIcs.STACS.2008.1322.

[38] Seiichiro Tani, Hirotada Kobayashi, and Keiji Matsumoto. Exact quantum algorithms for the leader election problem. *ACM Transactions on Computation Theory*, 4(1):1:1–1:24, 2012. doi:10.1145/2141938.2141939.

[39] Andrew Chi-Chih Yao. On the power of quantum fingerprinting. In *35th ACM Symposium on Theory of Computing (STOC)*, pages 77–81, 2003. doi:10.1145/780542.780554.